

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 27.08.2025 г.

Рабочая программа дисциплины

Профессиональный C++

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
01.03.02 - Прикладная математика и информатика

Направленность образовательной программы
Математическое моделирование и искусственный интеллект

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.ДВ.06.01 Профессиональный C++ относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ИП-2: Способен оптимизировать производительность программного обеспечения	ИП-2.1: Выполняет мониторинг производительности программного обеспечения ИП-2.2: Выполняет оптимизацию программного кода	ИП-2.1: ИП-2.1. 3-1. Знает методы и средства мониторинга производительности компьютерного программного обеспечения. ИП-2.1. 3-2. Знает метрики производительности программного обеспечения ИП-2.1. 3-3. Знает современные инструменты мониторинга производительности программного обеспечения. ИП-2.1. У-1. Умеет применять методы и средства мониторинга производительности компьютерного программного обеспечения. ИП-2.1. У-2. Умеет интерпретировать диагностические данные мониторинга производительности компьютерного программного обеспечения. ИП-2.1. У-3. Умеет определять "узкие места" программного обеспечения. ИП-2.2: ИП-2.2. 3-1. Знает методы и средства оптимизации производительности компьютерного программного	Собеседование	Зачёт: Контрольные вопросы

		<p>обеспечения.</p> <p>ИП-2.2. 3-2. Знает современные инструменты оптимизации производительности программного обеспечения</p> <p>ИП-2.2. У-1. Умеет оптимизировать программный код с использованием специализированных программных средств.</p> <p>ИП-2.2. У-2. Умеет вырабатывать варианты оптимизации производительности компьютерного программного обеспечения.</p> <p>ИП-2.2. У-3. Умеет проводить оценку и обоснование рекомендуемых решений по оптимизации производительности.</p>		
<p>ППК-Р6: Способен участвовать в промышленной разработке программного обеспечения (top)</p>	<p>ППК-Р6.1: Работает в соответствии с промышленными методологиями разработки</p> <p>ППК-Р6.2: Использует инструменты промышленной разработки</p> <p>ППК-Р6.3: Разрабатывает масштабируемый и поддерживаемый код</p>	<p>ППК-Р6.1:</p> <p>ППК-Р6.1. 3-1. Знает принципы Agile и их применение в промышленных проектах.</p> <p>ППК-Р6.1. 3-2. Знает процессы code review, принципы коллективного владения кодом (collective code ownership).</p> <p>ППК-Р6.1. У-1. Умеет оценивать объем задачи и срок ее выполнения, участвовать в планировании спринтов.</p> <p>ППК-Р6.1. У-2. Умеет работать в команде с использованием инструментов управления проектами.</p> <p>ППК-Р6.2:</p> <p>ППК-Р6.2. 3-1. Знает принципы Continuous Integration and Continuous Delivery (CI/CD).</p> <p>ППК-Р6.2. 3-2. Знает системы мониторинга и логирования в продуктивной среде.</p>	<p>Практическое задание</p>	<p>Зачёт: Контрольные вопросы</p>

		<p>ППК-Р6.2. У-1. Умеет настраивать потоки работ CI/CD.</p> <p>ППК-Р6.2. У-2. Умеет работать с контейнеризацией и оркестрацией.</p> <p>ППК-Р6.2. У-2. Умеет настраивать мониторинг в продуктивной среде.</p> <p>ППК-Р6.3:</p> <p>ППК-Р6.3. 3-1. Знает принципы чистого кода, SOLID, DRY, KISS и др.</p> <p>ППК-Р6.3. 3-2. Знает принципы предметно-ориентированного проектирования (ПОП) программного обеспечения.</p> <p>ППК-Р6.3. 3-3. Знает паттерны проектирования и антипаттерны.</p> <p>ППК-Р6.3. У-1. Умеет разрабатывать модульный и тестируемый программный код.</p> <p>ППК-Р6.3. У-2. Умеет выполнять модульное, интеграционное и нагрузочное тестирование.</p> <p>ППК-Р6.3. У-3. Умеет проводить рефакторинг для повышения качества кода.</p>		
--	--	--	--	--

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	2
Часов по учебному плану	72
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	16
- занятия семинарского типа (практические занятия / лабораторные работы)	16
- КСР	1
самостоятельная работа	39
Промежуточная аттестация	0 Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			Самостоятельная работа обучающегося, часы
		Контактная работа (работа во взаимодействии с преподавателем), часы из них		Всего	
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы		
0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	
Введение. Состав языка программирования C++. Эволюция C++.	8	2	2	4	4
Базовые концепции языка C++. Объявления и определения. Правило одного определения.	9	2	2	4	5
Строки в языке C++. С-строки, строки в виде класса. Подходы к повышению эффективности строк.	9	2	2	4	5
Шаблоны в C++. Шаблоны функций, вывод типов, шаблоны и перегрузка. Шаблоны классов. Двухфазное разрешение имен в шаблонах.	9	2	2	4	5
Механизм исключений в C++. Гарантии безопасности. Проектирование кода с учетом исключений..	9	2	2	4	5
Левые и правые ссылки в C++. Семантика перемещений.	9	2	2	4	5
Лямбда-выражения в C++. Замыкание и захват в лямбда-выражении.	9	2	2	4	5
Интеллектуальные указатели в C++. Идиома RAII. Уникальное и совместное владение ресурсами.	9	2	2	4	5
Аттестация	0				
КСР	1			1	
Итого	72	16	16	33	39

Содержание разделов и тем дисциплины

Цели и задачи изучения дисциплины.

Целью изучения дисциплины «Профессиональный C++» является освоение расширенных возможностей современных стандартов языка программирования C++. Изучаются типовые проблемы, возникающие в рамках индустриального программирования на C++, и методы их решения.

Задачами изучения дисциплины являются освоение ключевых возможностей C++: обобщенного программирования с использованием шаблонов, обработки ошибок с использованием механизма исключений, семантики перемещения на основе правых ссылок, элементов функционального программирования на основе лямбда-выражений, работы с памятью в семантике значений на основе интеллектуальных указателей.

1. Введение. Состав языка программирования C++. Эволюция C++.
2. Базовые концепции языка C++. Объявления и определения. Правило одного определения.
3. Строки в языке C++. С-строки, строки в виде класса. Подходы к повышению эффективности строк: подсчет ссылок, COW-строки и их плюсы и минусы, SSO-оптимизация строк. Полный вариант класса

string из стандартной библиотеки языка C++.

4. Шаблоны в C++. Шаблоны функций, вывод типов, шаблоны и перегрузка. Шаблоны классов.

Двухфазное разрешение имен в шаблонах.

5. Механизм исключений в C++. Обработка ошибок без исключений. Что является и что не является исключениями в стиле C++. Базовые концепции механизма исключений C++. Нейтральность кода относительно исключений. Гарантии безопасности. Проектирование кода с учетом исключений.

Примеры.

6. Левые и правые ссылки в C++. Ссылочное связывание. Семантика перемещений. Особенности реализации конструкторов копирования и перемещения, копирующего и перемещающего присваивания.

Правило пяти. Правило нуля.

7. Виды функторов в языке C++. Лямбда-выражения в C++. Синтаксис, особенности объявления и использования. Замыкание и захват в лямбда-выражении. К чему можно привести лямбда-выражение. Рекурсии в лямбда-выражениях.

8. Интеллектуальные указатели в C++. Идиома RAII. Уникальное и совместное владение ресурсами.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

1. Bjarne Stroustrup. A Tour of C++, 3rd Edition. – Addison-Wesley Professional, 2022.
2. Бьерн Страуструп. Язык программирования C++. Специальное издание. – М.: Бином, 2011.
3. Николаи Джосаттис: Стандартная библиотека C++: справочное руководство, 2-е изд. – М.: Вильямс, 2014. – 1136 с.
4. Герб Саттер. Новые сложные задачи на C++. Серия "C++ In-Depth". – М.: Вильямс, 2015. – 272 с.
5. Герб Саттер. Решение сложных задач на C++. Серия "C++ In-Depth" – М.: Вильямс, 2015. – 400 с.
6. Мейерс С. Эффективный и современный C++: 42 рекомендации по использованию C++11 и C++14. – М.: Вильямс, 2016.

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Собеседование) для оценки сформированности компетенции ИП-2:

1. Какие существуют контейнеры STL? Каковы сложность операций работы с ними?

Примеры.

2. Типы итераторов. Связь контейнеров STL и итераторов.
3. Примеры использования decltype и auto, основные правила вывода типов.

4. Почему невозможно копировать объект типа `std::unique_ptr`? Примеры.
5. Что такое `std::weak_ptr`? Зачем он нужен? Примеры.
6. Типичные шаблоны проектирования с использованием интеллектуальных указателей.

Критерии оценивания (оценочное средство - Собеседование)

Оценка	Критерии оценивания
зачтено	Студент дал развернутый ответ на все вопросы без существенных ошибок.
не зачтено	При ответе студент допускает грубые ошибки в основном материале.

5.1.2 Типовые задания (оценочное средство - Практическое задание) для оценки сформированности компетенции ППК-Р6:

1. Освоение использования контейнеров STL на типовых задачах.
2. Освоение работы с итераторами как с основой доступа к элементам контейнера.
3. Освоение механизма вывода типов C++ и лучших практик его использования.
4. Освоение использования интеллектуальных указателей и типовых вариантов их применения.

Критерии оценивания (оценочное средство - Практическое задание)

Оценка	Критерии оценивания
зачтено	Студент дал развернутый ответ на все вопросы без существенных ошибок.
не зачтено	При ответе студент допускает грубые ошибки в основном материале.

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено			зачтено			
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много	Уровень знаний в объеме, соответствующем программе подготовки	Уровень знаний в объеме, соответствующем программе подготовки	Уровень знаний в объеме, соответствующем программ	Уровень знаний в объеме, превышающем программу подготовки.

	вследствие отказа обучающегося от ответа		негрубых ошибок	. Допущено несколько негрубых ошибок	. Допущено несколько несущественных ошибок	и. Ошибок нет.	
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторым и недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторым и недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой
	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ИП-2

1. Какие существуют контейнеры STL? Каковы сложность операций работы с ними?

Примеры.

2. Типы итераторов. Связь контейнеров STL и итераторов.
3. Примеры использования decltupe и auto, основные правила вывода типов.

5.3.2 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ППК-Р6

1. Почему невозможно копировать объект типа `std::unique_ptr`? Примеры.
2. Что такое `std::weak_ptr`? Зачем он нужен? Примеры.
3. Типичные шаблоны проектирования с использованием интеллектуальных указателей.

Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	Студент ответил на большую часть вопросов возможно с незначительными недочетами.
не зачтено	При ответе студент допускает грубые ошибки в основном материале и решении стандартных задач.

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Язык программирования C++ для профессионалов / Страуструп Б. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663690&idb=0>.

Дополнительная литература:

1. Страуструп Б. Дизайн и эволюция C++ : монография / Страуструп Б. - Москва : ДМК-пресс, 2016. - 446 с. - ISBN 978-5-97060-419-9., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=868528&idb=0>.
2. Подбельский В.В. Язык Си++ : учебное пособие / Подбельский В.В. - Москва : Финансы и статистика, 2022. - 560 с. - ISBN 978-5-00184-082-4., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=869552&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

Интернет-ресурсы:

<https://e-lib.unn.ru/>

<https://intuit.ru/>

Программное обеспечение:

Программное обеспечение:

1. Операционные системы семейства Microsoft Windows, лицензия по подписке Microsoft Imagine.
2. Браузер Google Chrome, предоставляется бесплатно на условиях лицензионных соглашений на программное обеспечение с открытым исходным кодом.
3. Среда разработки семейства Microsoft Visual Studio <https://visualstudio.microsoft.com/ru/>

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 01.03.02 - Прикладная математика и информатика.

Автор(ы): Сысоев Александр Владимирович, кандидат технических наук, доцент.

Заведующий кафедрой: Мееров Иосиф Борисович, кандидат технических наук.

Программа одобрена на заседании методической комиссии от 25.06.2025, протокол № Протокол №11.